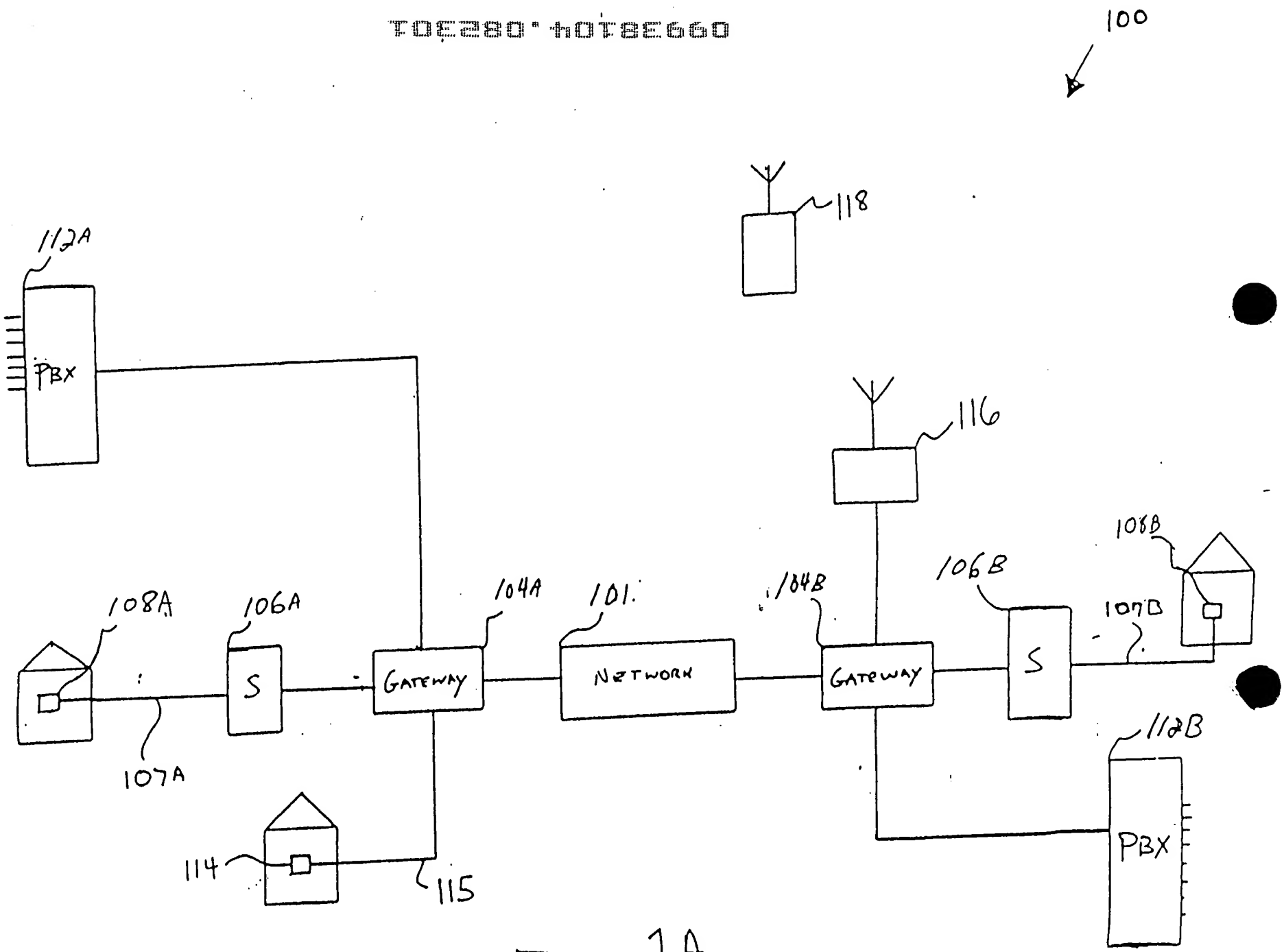


FIG. 1A

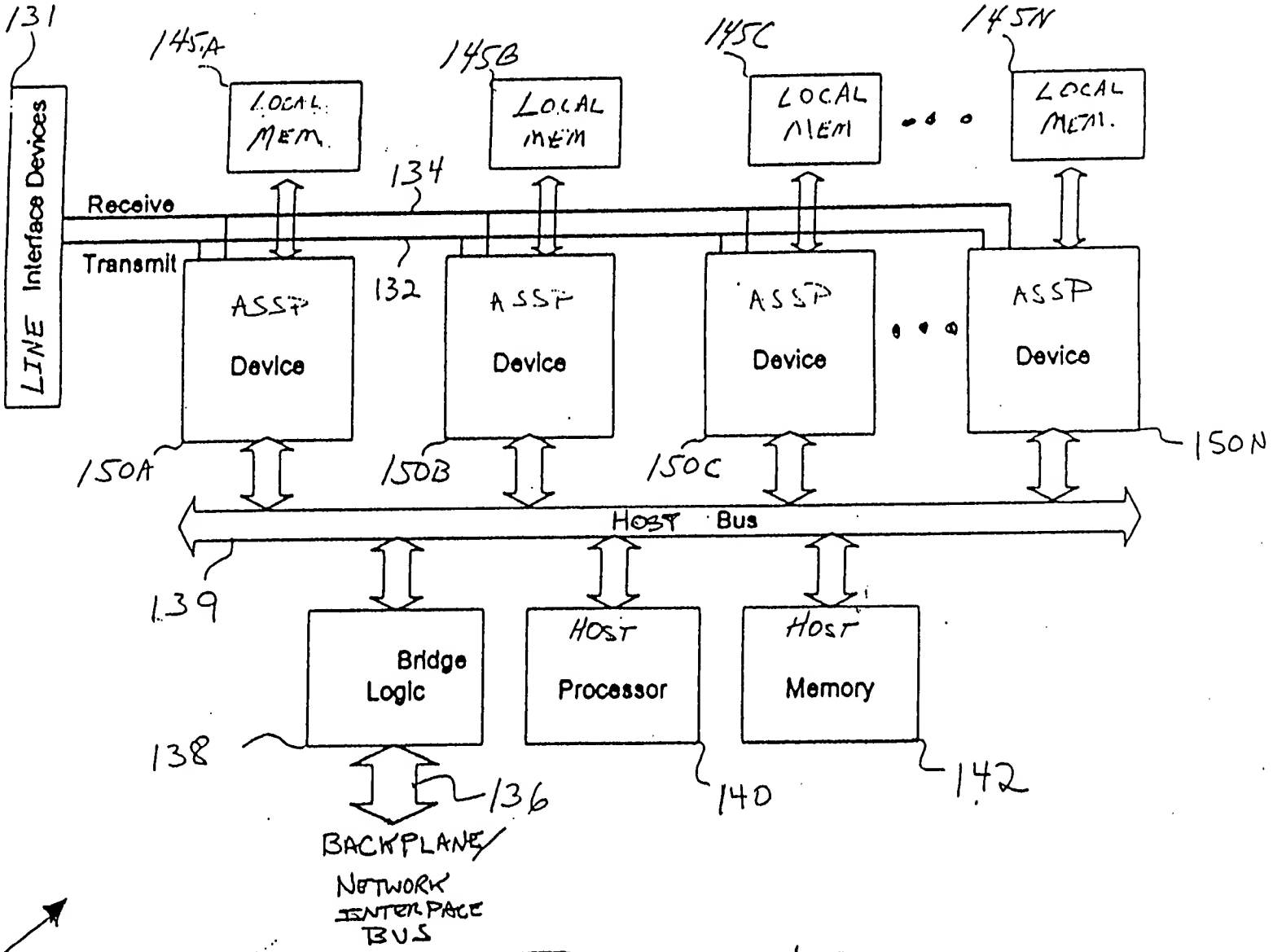

131
LINE Interface Devices
145A
LOCAL MEM.
145B
LOCAL MEM
145C
LOCAL MEM
145N
LOCAL MEM.
134
Receive
Transmit
132
ASSP Device
ASSP Device
ASSP Device
ASSP Device
150A
150B
150C
150N
Host Bus
139
Bridge Logic
Host Processor
Host Memory
138
136
140
142
BACKPLANE/ NETWORK INTERFACE BUS
130

FIG. 1B

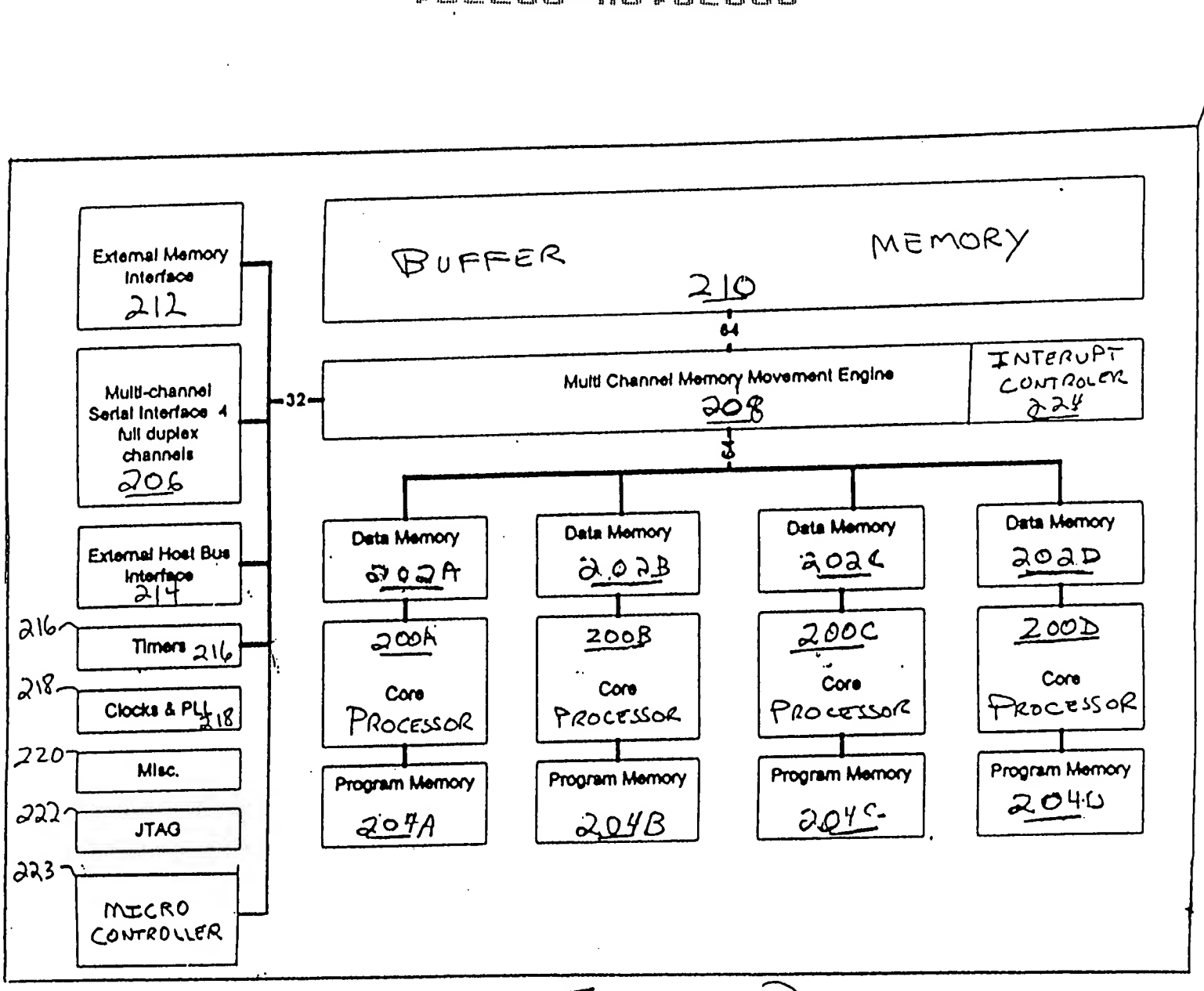


FIG. 2

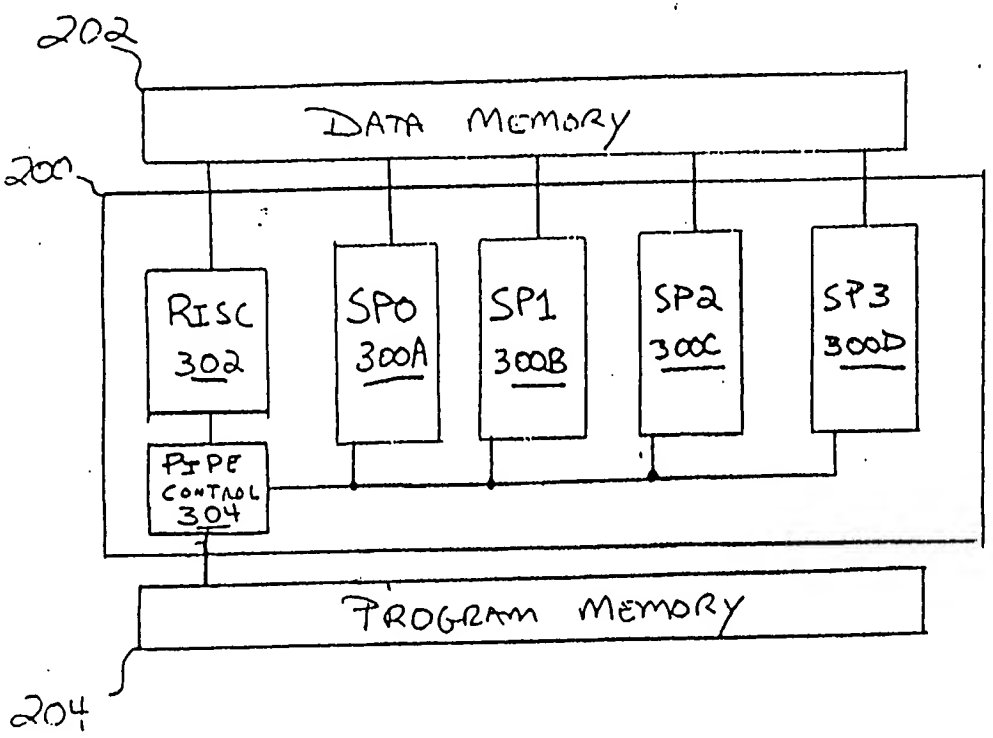


FIG. 3

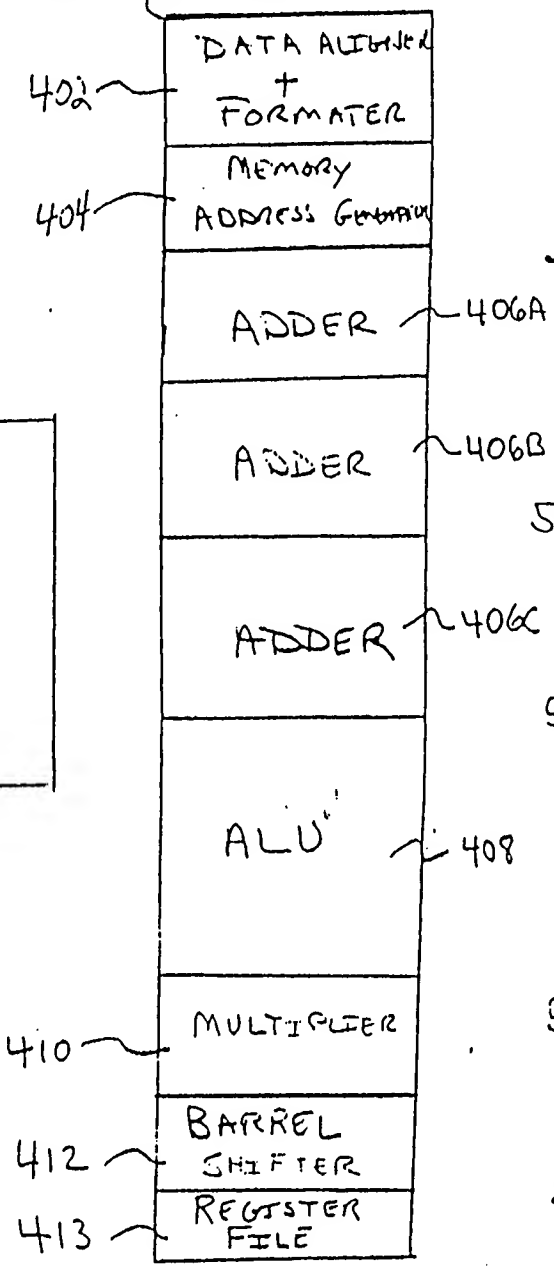


FIG. 4

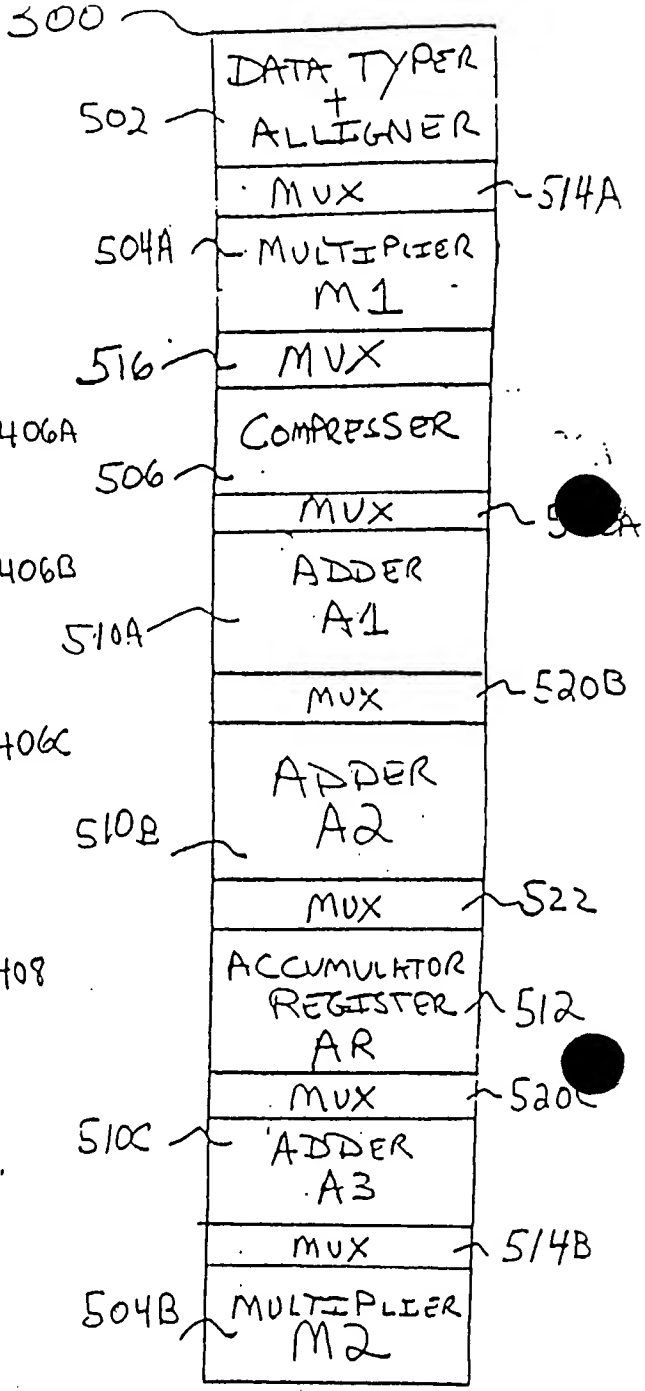


FIG. 5A

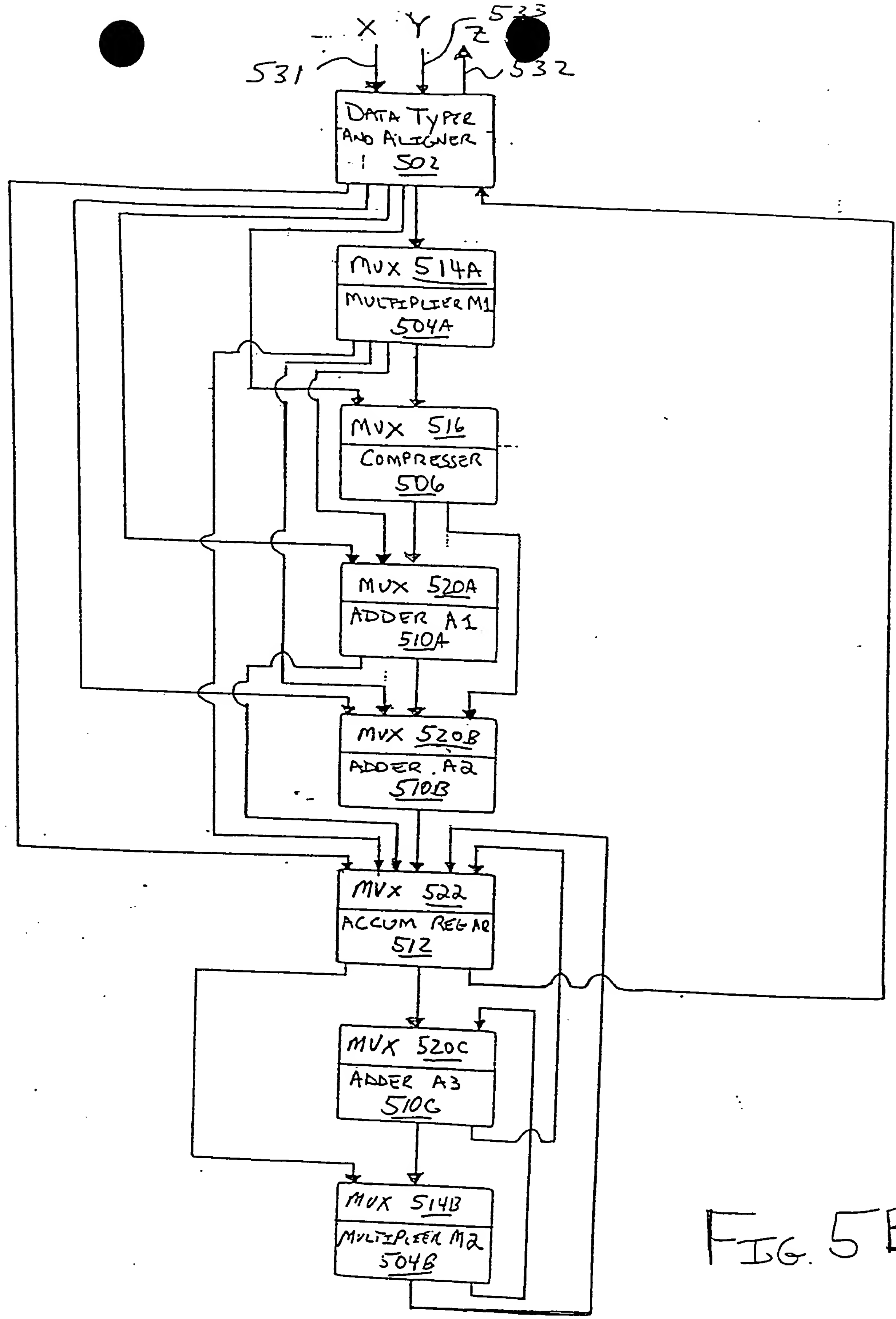


FIG. 5B

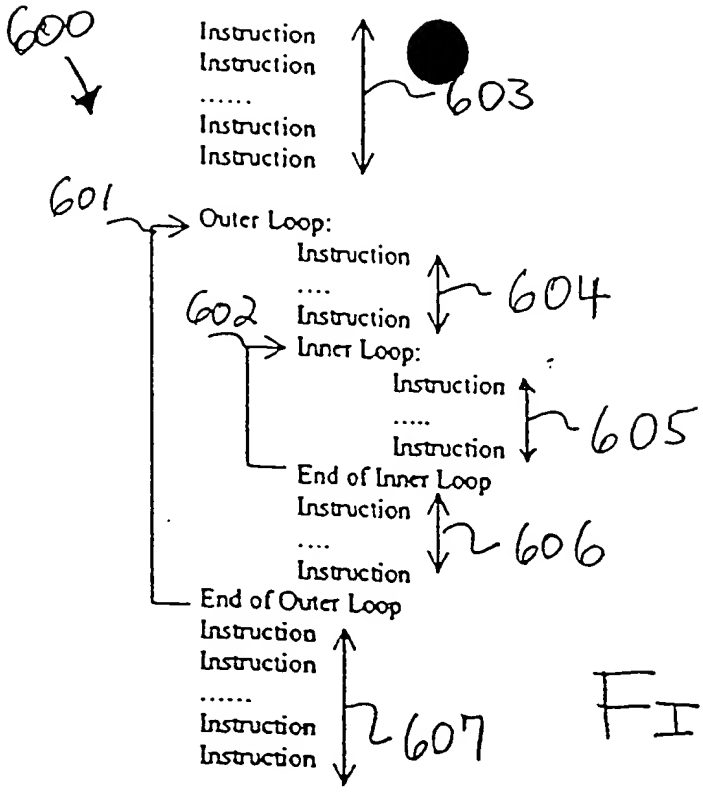


FIG. 6A

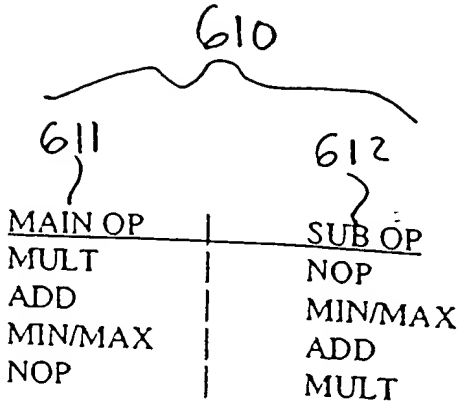


FIG. 6B

FIG. 6C

39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	0	PS	S"	SX					SY				V/S	SA	DA	Sub-op			1	Pred		PL		Sxt		Syt		Rnd		S"	S"	S"	0	SA	DA	abs	0	0

da = +/- sx*sy	Nop	0	0	0
da = +/-(sx*sy) + sa	Add	0	0	1
da = +/-(sx*sa) + sy	Add	0	1	0
da = +/-(sx*sy) - sa	Sub	0	1	1
da = +/-(sx*sa) - sy	Sub	1	0	0
da = min(+/- sx*sy,sa)	Min	1	0	1
da = min(+/- sx*sa,sy)	Min	1	1	0
da = max(+/- sx * sy, sa)	Max	1	1	1

Lt			
Lt			
Lt			
Lt			
Gx			
Gx			
Gx			

FIG. 6D

39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20		
1	0	0	PS		S"	SX				SY				V/S	SA	DA	0	1	0	Add	da = +/-(mx*sa) + my
																	1	0	0	Sub	da = +/-(mx*sa) - my
																	1	1	0	Min	da = min(+/-mx*sa, my)

20-bit ISA

39	19	
0	0	20-bit parallel
0	1	20-bit serial
1	0	40-bit extended
1	1	20-bit serial

Control || Control
Control # Control
DSP, extensions/Shadow
DSP # DSP

DSP Instructions

	39	38	37	36	35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20
Multiply	1	0	0	PS	S*	SX					SY				V/S	SA	DA	Sub-op		
da = sx*sy																		0	0	0 Nop
da = (sx*sy) + sa																		0	0	1 Add
da = (sx*sa) + sy																		0	1	0 Add
da = (sx*sy) - sa																		0	1	1 Sub
da = (sx*sa) - sy																		1	0	0 Sub
da = min(sx*sy,sa)																		1	0	1 Min
da = min(sx*sa,sy)																		1	1	0 Min
da = max(sx*sy,sa)																		1	1	1 Max
Add	1	0	1	PS	+/-	SX					SY				V/S	SA	DA	Sub-op		
da = sx + sy																		0	0	0 Nop
da = sx + sy + sa																		0	0	1 Add
da = sx + sy; sa = sx - sy;																		0	1	0 AddSub
da = (sx + sy) * sa																		0	1	1 Mul
da = -(sx + sy) * sa																		1	0	0 MulN
da = min(sx+sy,sa)																		1	0	1 Min
da = max(sx+sy,sa)																		1	1	0 Max
da = ssum(sa) (sx, sy unused)																		1	1	1 CombAdd
Extremum	1	1	0	PS	X/N	SX					SY				V/S	SA	DA	Sub-op		
da = ext(sx,sy)																		0	0	0 Nop
da = ext(sx,sy,sa)																		0	0	1 Ext
da = ext(sx,sa) * sy																		0	1	0 Mul
da = -ext(sx,sa) * sy																		0	1	1 MulN
da = ext(sx,sa) + sy																		1	0	0 Add
da = ext(sx,sa) - sy																		1	0	1 Sub
ext(sa,da) ? t = sx, tr = sy, tcs = tc																		1	1	0 amax
Type-match Permute	1	1	0	PS	0	SX					SY				x	x	x	1	1	1
	1	1	0	PS	1	SX					Type				x	ereg		1	1	1 Permute
rved	1	1	1	PS	x	SX					SY				SA	DA	V/S	Sub-op		

Control and specifier Extensions

	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Mul	0	Pred	PL		Sx1		Sy1		Rnd			S*	S*	S*	0	SA	DA	abs	0	0
											Lt									Add/Sub
											Gx									min/max
Add	0	Pred	PL		Sx1		Sy1		Lt		Sub-ext				0	SA	DA	abs	0	0
											+/-	+/-	+/-	x						Nop (uadd)
											x	V/S	Rnd	Fp						Mul/MulN
											tr-ctl		Gx	Fp						Min/max
Ext	0	Pred	PL		Sx1		Sy1		tr-ctl	Gx	Sub-ext				0	SA	DA	abs	0	0
											Lt	Fp								Add/sub
											Rnd	V/S								Mul
	0	Pred	PL		Sx1		Pctl							0	ereg		Pctl	0	0	

Type/offset/permute extensions

	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	Pred	PL		x	Type:SX					Type:SY				0	SA	DA	x	0	1 Type override
	0	Pred	PL		Psx	Permute:SX					Permute:SY				0	SA	DA	Psy	1	0 permute override
	0	Pred	I/R	I/R	prx	Offset:SX					Offset:SY				0	SA	DA	pry	1	1 Offset override

Shadow DSP

	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	Op	PL		op	ereg					ereg				1	SA	DA	Sub-op		

FIG. 6 E

Control Instructions

	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
add,sub	L	Pred	0	0	0	RX					RY				RZ				+/-	0
max,mun	L	Pred	0	0	0	RX					RY				RZ				X/N	1
Shift	L	Pred	0	0	1	RX					UI4				RZ				UI1	R/L
Logic	L	Pred	0	1	0	RX					RY				RZ				&, |	&|
Mux	L	Pred	0	1	1	RX					RY				RZ				Pd	0
mov	L	Pred	0	1	1	RX					DZ				Rxt	Dzt	0	0	0	1
addi	L	Pred	0	1	1	SI4					DZ				x	x	1	0	0	1
mov2erg	L	Pred	0	1	1	RX					unit			ereg	gd	type			1	0	1
Ldm	L	Pred	0	1	1	RX					DZ1				DZ2				1	1
‘tbits	L	Pred	1	0	0	UI4:POS					RZ				Rzt	UI4				0
bits	L	Pred	1	0	0	UI4:POS					RZ				Rzt	UI2		0	0	1
Setbit	L	Pred	1	0	0	UI4:POS					RZ				Rzt	UI1	UI1	1	0	1
Movl	L	Pred	1	0	0	SI8									RZ				·1	1
Jmp	L	Pred	1	0	1	SI9										0	Pred		0	0
Call	L	Pred	1	0	1	SI9										1	Pred		0	0
Loop	L	Pred	1	0	1	UI5 ; Lcount					UI5: Lsize							UI2; Lst	0	1
Jmpi	L	Pred	1	0	1	RX	x	x	x	x	x	x	0	Pred					1	0
Calli	L	Pred	1	0	1	RX	x	x	x	x	x	x	1	Pred					1	0
Loopi	L	Pred	1	0	1	RX	x	UI5: Lsize										UI2: Lst	1	1
Tst	L	Pred	1	1	0	RX					RY				PZ				=, <, >	0
Testbit	L	Pred	1	1	0	RX					UI5				PZ			B	0	1
Andp, orp	L	Pred	1	1	0	Pa			Pb			Pc			PZ			&|	1	1
Load	L	Pred	1	1	1	MX					RZ				Ext			0	0	0
Store	L	Pred	1	1	1	MZ					RX				Ext			1	0	0
eLoad	L	Pred	1	1	1	MX					RZ				1	1	1	0	0	0
eStore	L	Pred	1	1	1	MZ					RX				1	1	1	1	0	0
Extended	L	Pred	1	1	1	Bits 27:16													1	0
Logic2	L	Pred	1	1	1	RX					RY/RZ				Rxt	Ryt	&, |, &|, |		0	1
mov-erg	L	Pred	1	1	1	unit		ereg			RZ				gd	Sft		0	1	1
Crc	L	Pred	1	1	1	RX					RZ				s/m	0	0	1	1	1
Parity	L	Pred	1	1	1	RX					PZ			O/E	0	1	0	1	1	1
Stm	L	Pred	1	1	1	MZ					RX				1	1	0	1	1	1
Abs	L	Pred	1	1	1	RX					RZ				0	0	1	1	1	1
Neg	L	Pred	1	1	1	RX					RZ				0	1	1	1	1	1
Div-step	L	Pred	1	1	1	RX					RZ				1	0	1	1	1	1
& Set	L	Pred	1	1	1	RX					PZ			0	1	1	1	1	1	1
[illegible]	[illegible]																			
Return	L	Pred	1	1	1	Pred		t-cu			0	1	0	1	1	1	1	1	1	1
Zero-ac	L	Pred	1	1	1	ac #					1	1	0	1	1	1	1	1	1	1
eSync	L	Pred	1	1	1	RZ					0	1	1	1	1	1	1	1	1	1
Swi	L	Pred	1	1	1	UI3				0	1	1	1	1	1	1	1	1	1	1
Nop	L	Pred	1	1	1	UI3				1	1	1	1	1	1	1	1	1	1	1

<Bit1, Bits9-6> == UI5 (Shift Amount)

<Bit3, Bits13-10> == UI5 :POS

FIG. 6 F

Extended Control

	Bits 13:2 of upper half (39:20)																												
	13	12	11	10	9	8	7	6	5	4	3	2	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Insert/Extract	RX				RZ				0	0	0	0	0	x	x	0	Rxt	Rzt	I/E	RA	RA	Offset:UI5					Length:UI5					0
																						x	RY				RV				x	
Inserti	UI4: length				RZ				0	0	0	1	0	x	x	0	Rzt	UI5: Position				Imm10										
Shift	RX				RZ				0	0	0	0	0	rxh	rxl	0	rzh	rzl	D	U/S	1	Shift: UI5					A/L	LI	R/L	0	Fill	1
																					0	ryh	RY									
Rotate	RX				RZ				0	0	0	0	0	x	x	0	x	x	x	x	1	Shift: UI5					1	x	R/L	1	x	1
																					0	ryh	RY									
jmp, call	UI7							J/C	0	0	1	0	0	Pred		0	x	UI15														
dloop	UI4: outer LC				UI4: inner LC				0	0	1	1	0	x	exII	0	UI1	UI4: outer Lsize				UI4: inner Lsize				UI2: O-Ls	UI4: inner Lstart				0
dloopi	RX				RY				0	0	1	1	0	x	exII	0	x	UI4: outer Lsize				UI4: inner Lsize				UI2: O-Ls	UI4: inner Lstart				1
¶	RX				RY				0	1	0	0	0	x	x	0	0	rxh	md	ryh	+/-	e/+	RZ				I/I	rzh	rzl	s/u	s/u	0
sub	RX				RY				0	1	0	0	0	x	x	0	0	rxh	rxl	ryh	ryl	+/-	RZ				LI	rzh	rzl	x	x	1
[illegible]	[illegible]																															
logicp	Px			D	PZ				0	1	0	0	0	x	x	0	1	T/F	T/F	T/F	&I	&I	PY				PV				x	1
Testi	RX			D	PZ				0	1	0	1	0	=,>,<		0	Imm16															
Movi	H/L	Fill	RZ						0	1	1	0	0	x	x	0	Imm16															
loadi	Type		RZ						0	1	1	1	0	x	x	0	0	0	Imm14														
storei	Type		RX						0	1	1	1	0	x	x	0	0	1	Imm14														
loadi	RX		RZ						0	1	1	1	0	x	x	0	1	Rzt	0	Type	SI10												
storei	MZ		RX						0	1	1	1	0	x	x	0	1	Rzt	1	Type	SI10												
Addi/subi	RX		RZ						1	0	+/-	0	0	LI	s/u	0	Imm16															
mini,maxi	RX		RZ						1	0	X/N	1	0	x	x	0	Imm16															
andi,ori	RX		RZ						1	1	&I	H/L	0	x	x	0	Imm16															

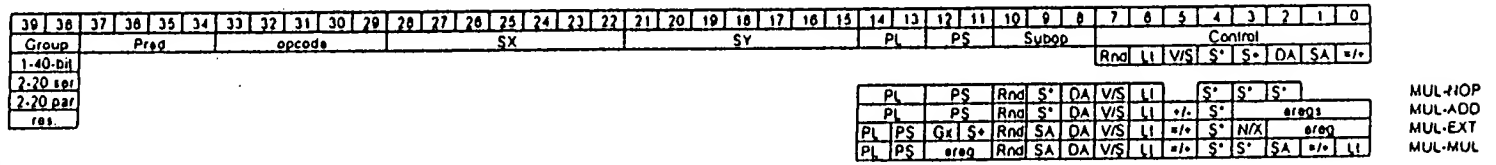
Fill: Sign/Zero

Bit 15 is continuation of inner LC

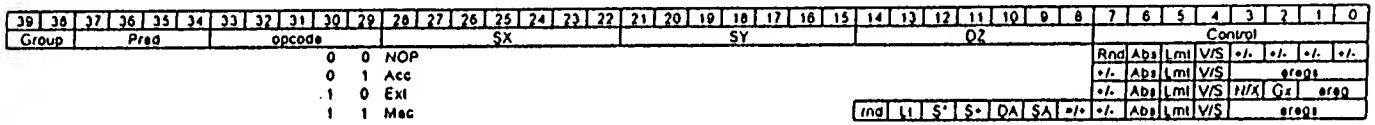
andp, orp, andorp, orandp: pz = (px relop py) relop pv

FIG. 6G

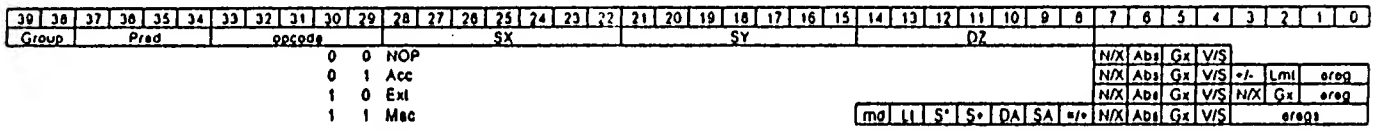
MAC:



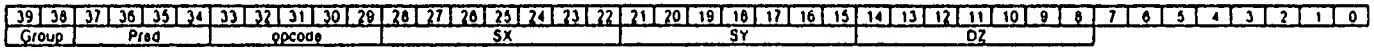
ARITH:



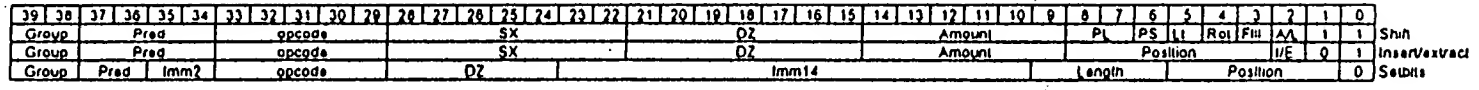
EXT:



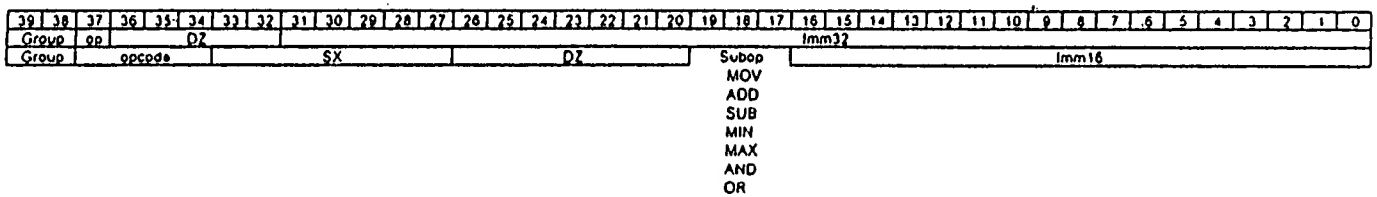
LOGIC:



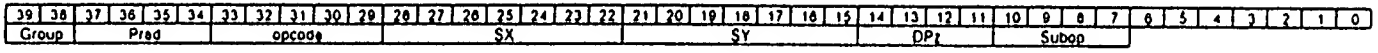
SHIFT:



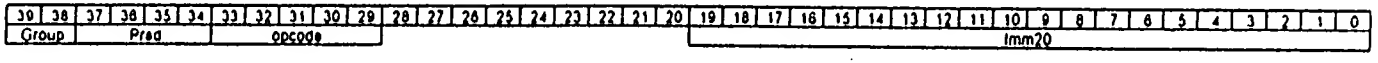
Immediate:



Test:



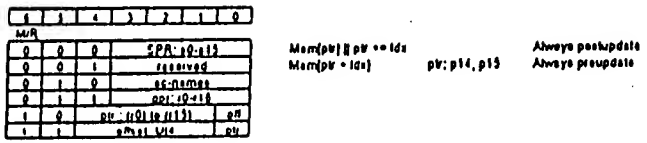
Branch:



Misc:

FIG. 6H

7-bit specifier: Parallel Store, Parallel Load in DSP Instructions



6-bit specifier: DSP instructions

5-bit specifier: RISC instructions

4-bit specifier:

AR

as-names:

3	2	1	0		SPR:
0	0	0	0	A0 (use type, SIMD)	gpr-type
0	0	0	1	A1	areg-type
0	0	1	0	T	lc-ctl
0	0	1	1	TR	pio-ctl
0	1	0	0	A00 (unit 0)	cb-ctl
0	1	0	1	A10	loop-ctl
0	1	1	0	T0	psr
0	1	1	1	TR0	status
1	0	0	0	SX1	
1	0	0	1	SX1s	
1	0	1	0	SX2	
1	0	1	1	SX2s	
1	1	0	0	SY1	
1	1	0	1	SY1s	
1	1	1	0	SY2	
1	1	1	1	SY2s	

areg-names

3	2	1	0	
0	0	0	0	A0
0	0	0	1	A1
0	0	1	0	T
0	0	1	1	TR
0	1	0	0	PP0
0	1	0	1	Aout
0	1	1	0	PP1
0	1	1	1	Dout
1	0	0	0	SX1
1	0	0	1	SX1s
1	0	1	0	SX2
1	0	1	1	SX2s
1	1	0	0	SY1
1	1	0	1	SY1s
1	1	1	0	SY2
1	1	1	1	SY2s

FIG. 6I

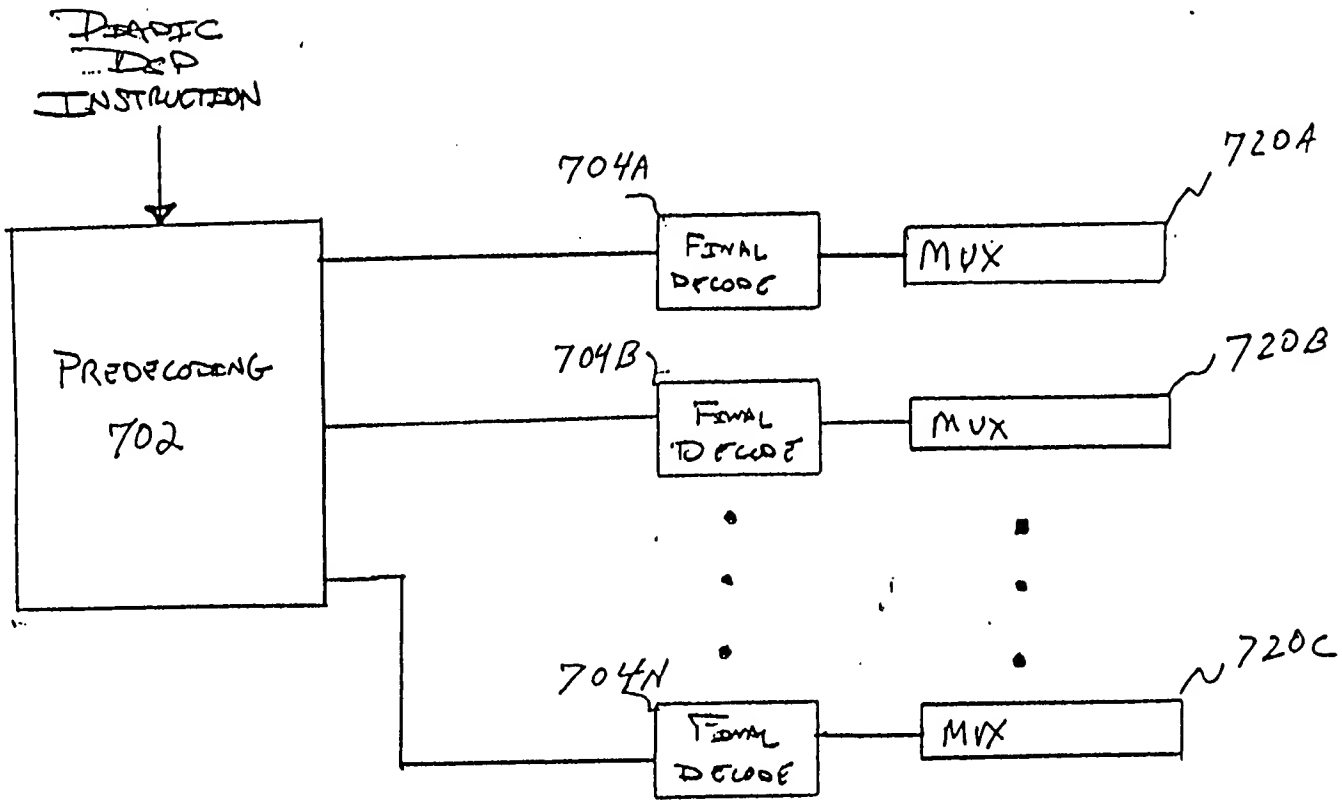


FIG. 7

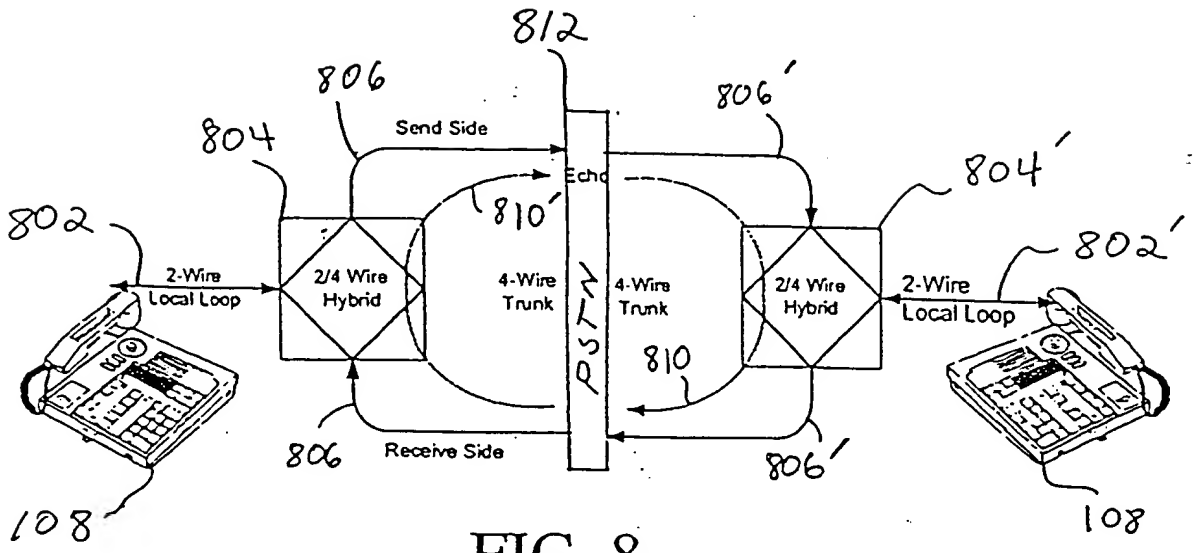


FIG. 8
(PRIOR ART)

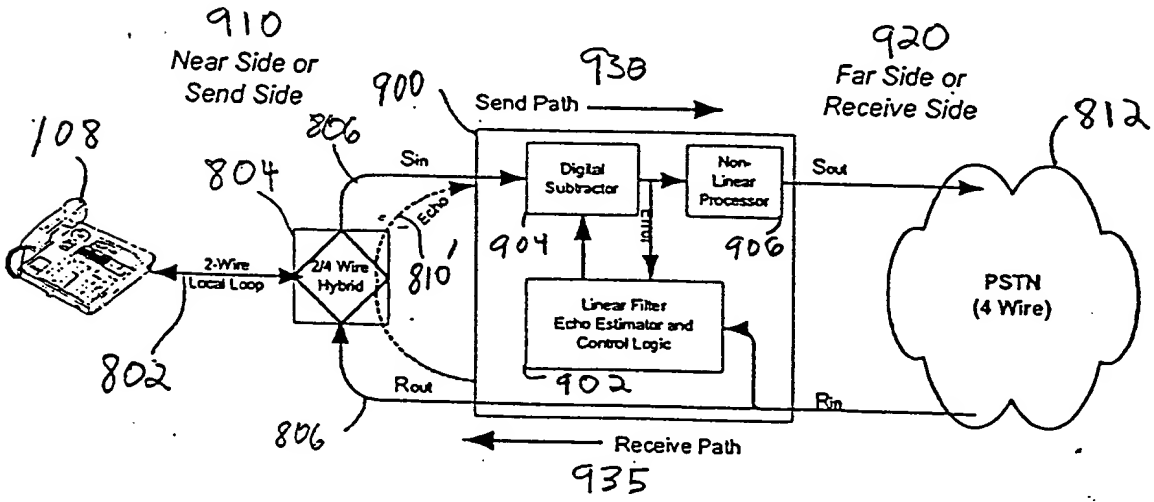


FIG. 9
(PRIOR ART)

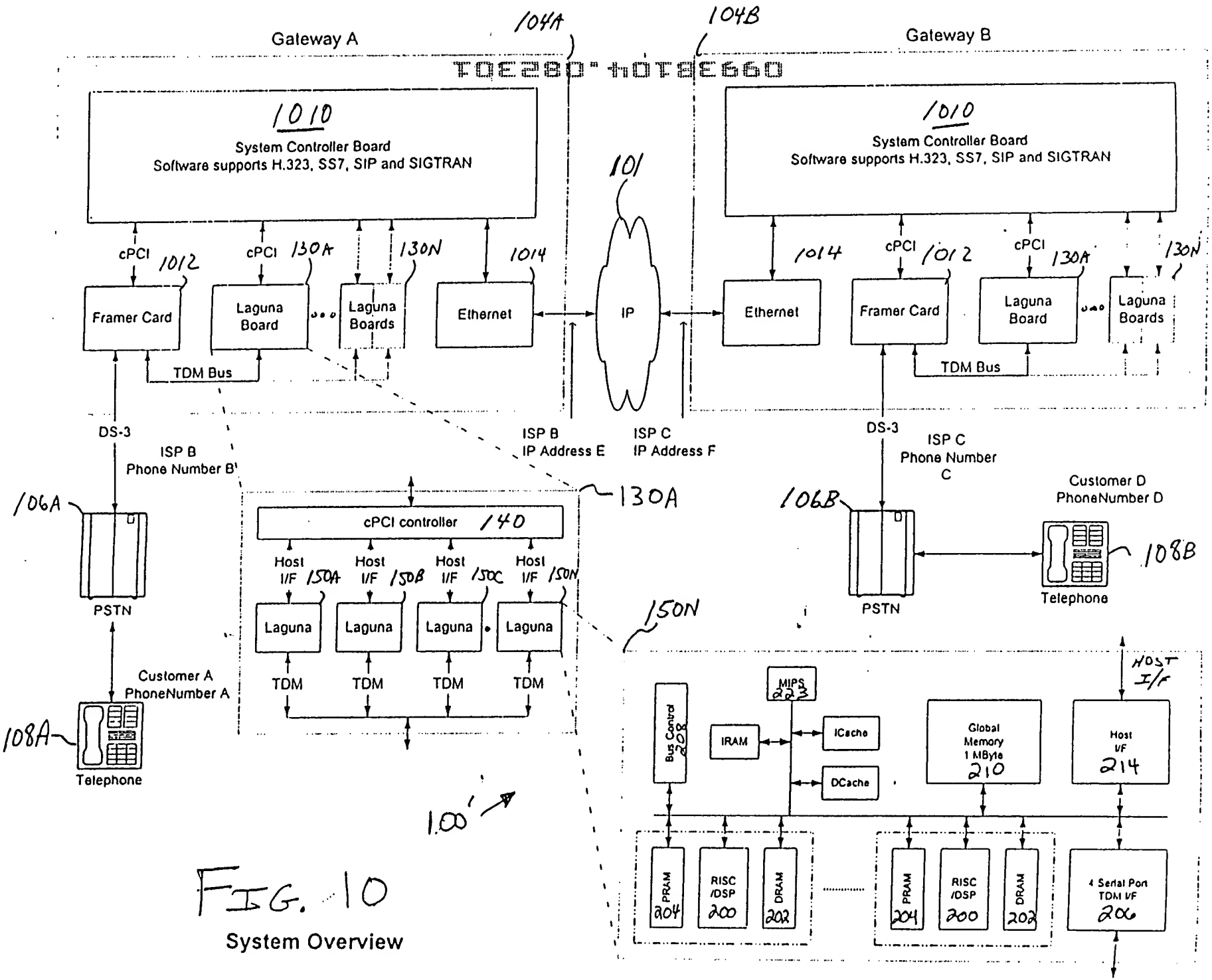


FIG. 10

System Overview

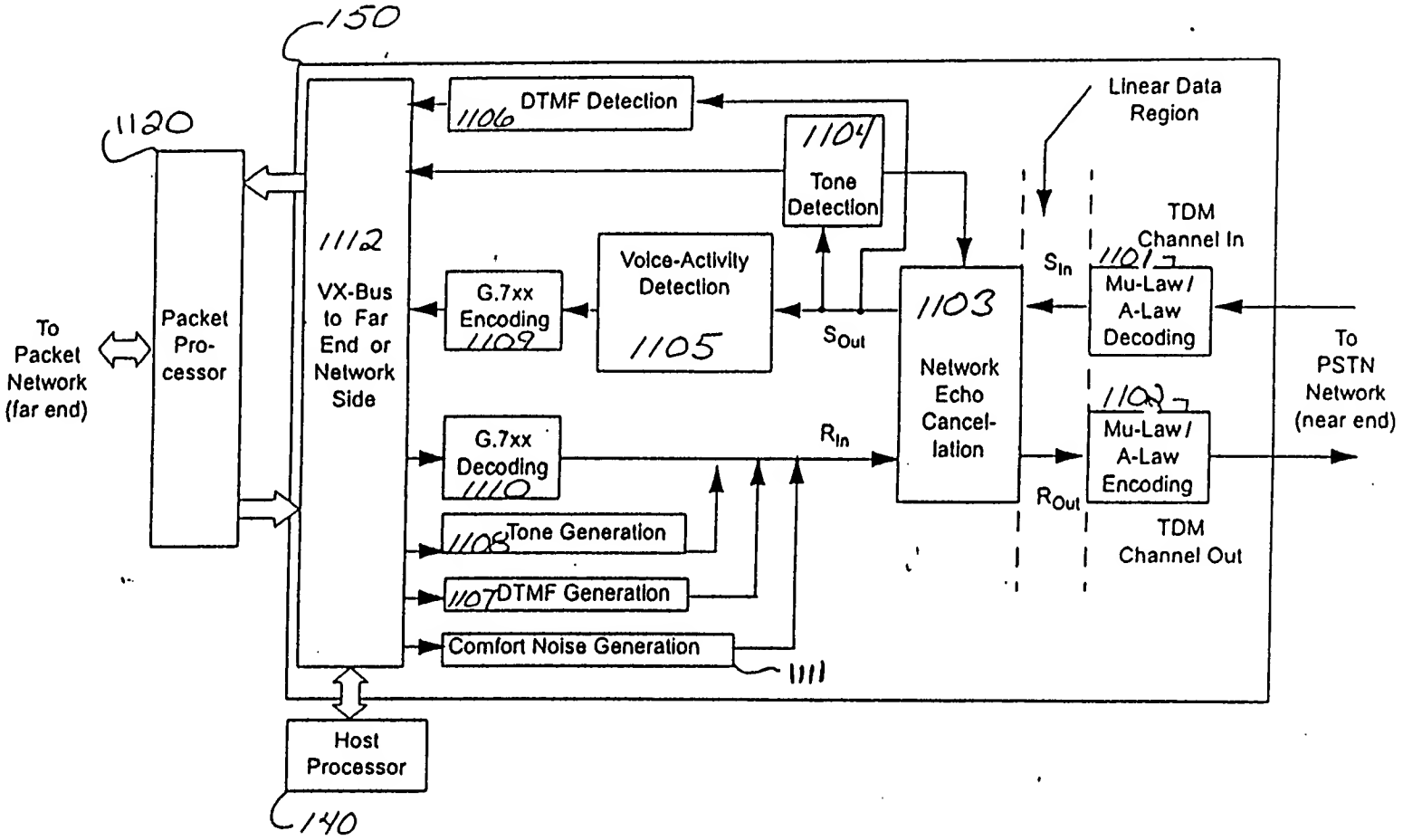


FIG. 11

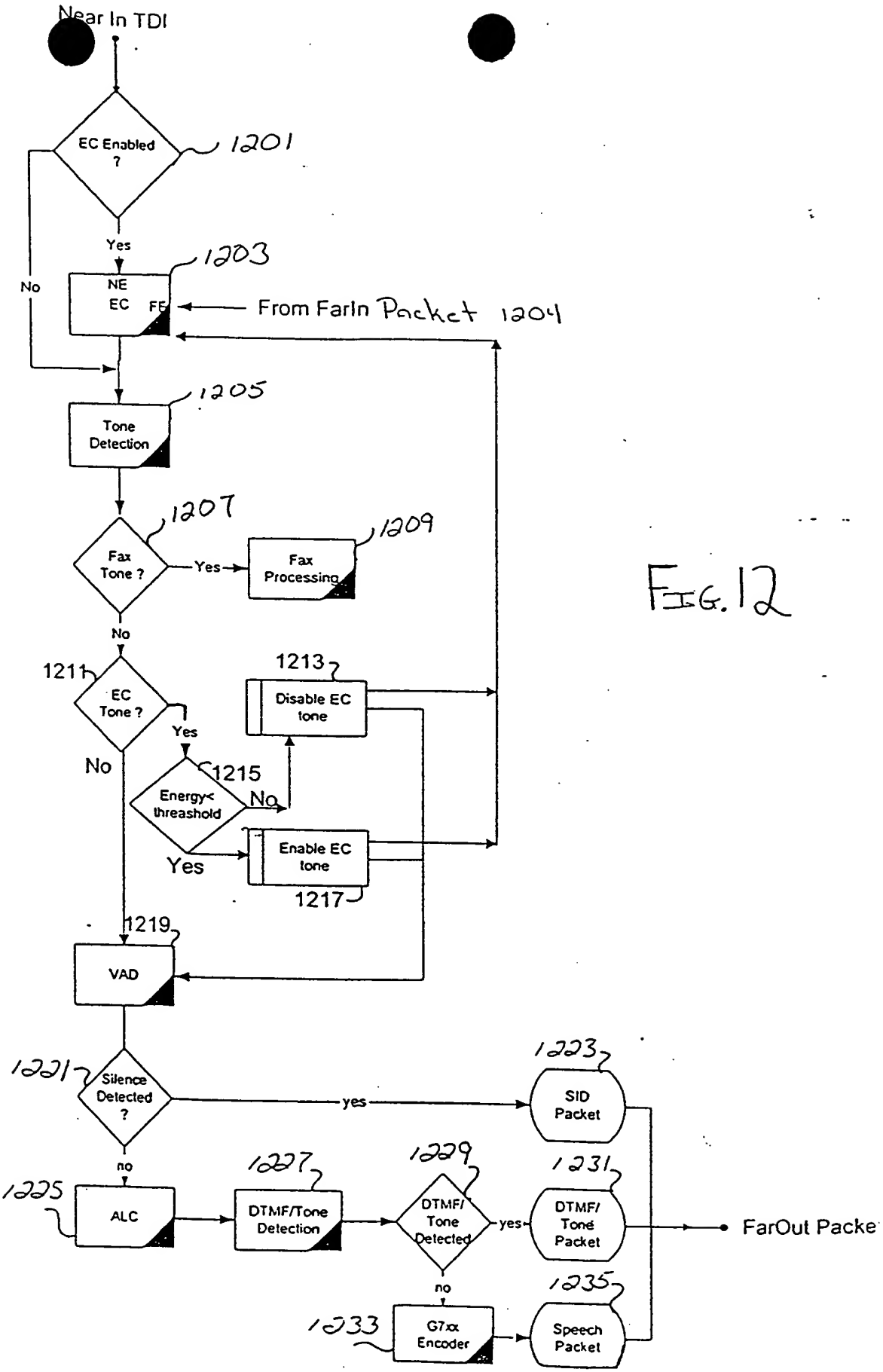

Near In TDI
EC Enabled ?
1201
Yes
No
1203
NE EC FE
From FarIn Packet 1204
1205
Tone Detection
1207
Fax Tone ?
Yes
Fax Processing
1209
No
1211
EC Tone ?
Yes
No
1213
Disable EC tone
1215
Energy< threashold
No
Yes
Enable EC tone
1217
1219
VAD
1221
Silence Detected ?
yes
1223
SID Packet
no
1225
ALC
1227
DTMF/Tone Detection
1229
DTMF/ Tone Detected
yes
1231
DTMF/ Tone Packet
FarOut Packe
no
1233
G7xx Encoder
1235
Speech Packet

FIG. 12

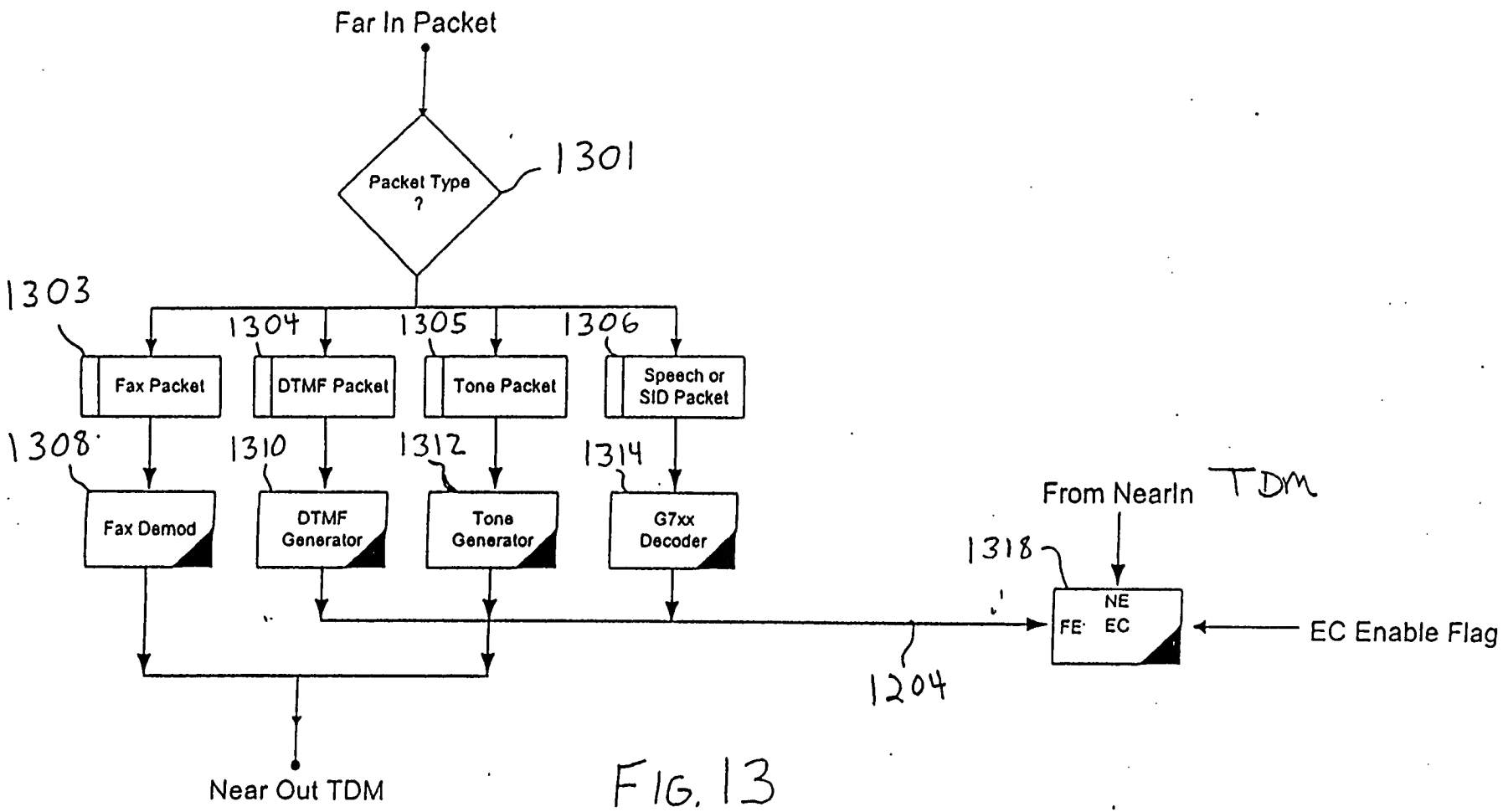


FIG. 13

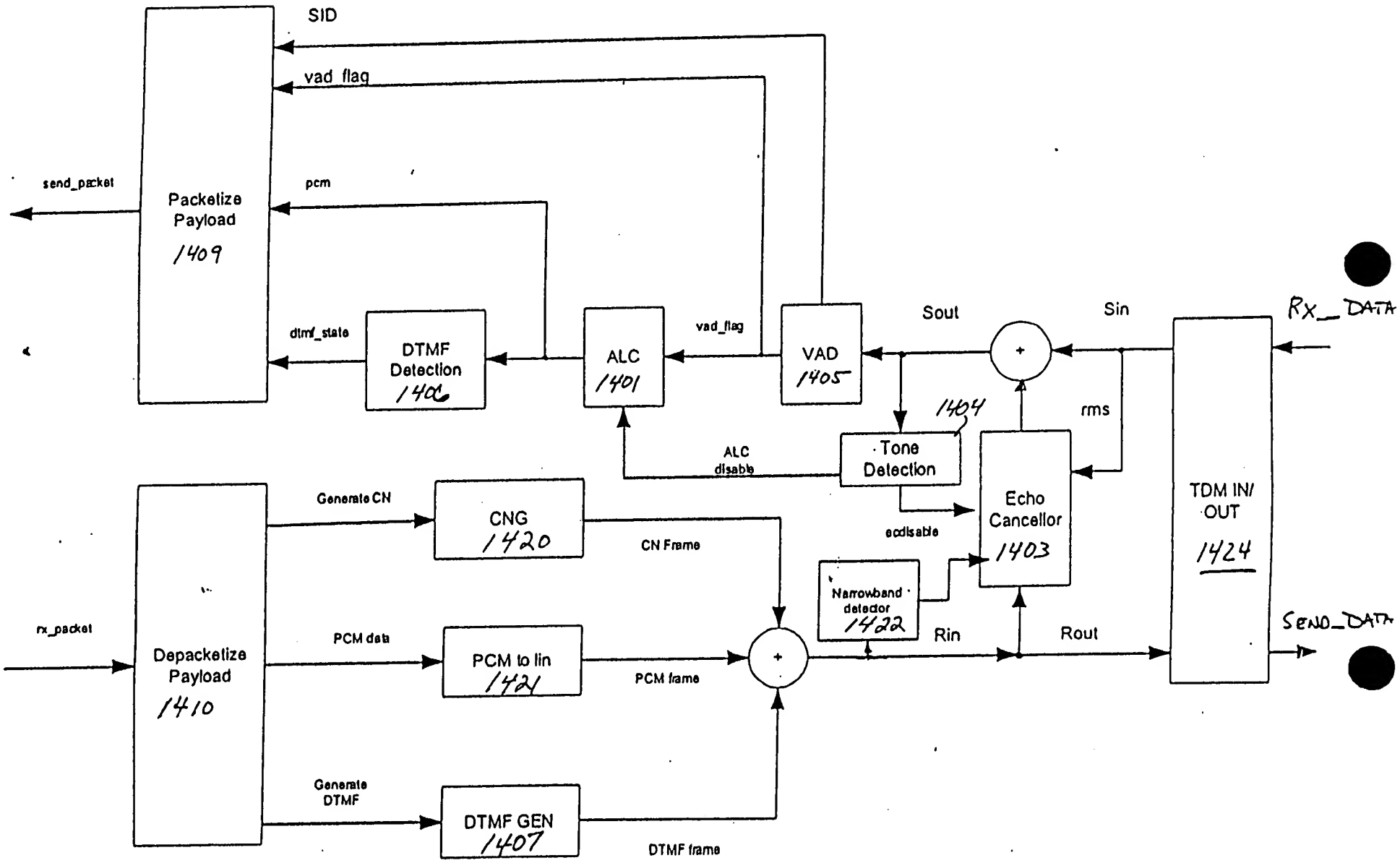


FIG. 14A

VxTel Voice Activity Detection Algorithm

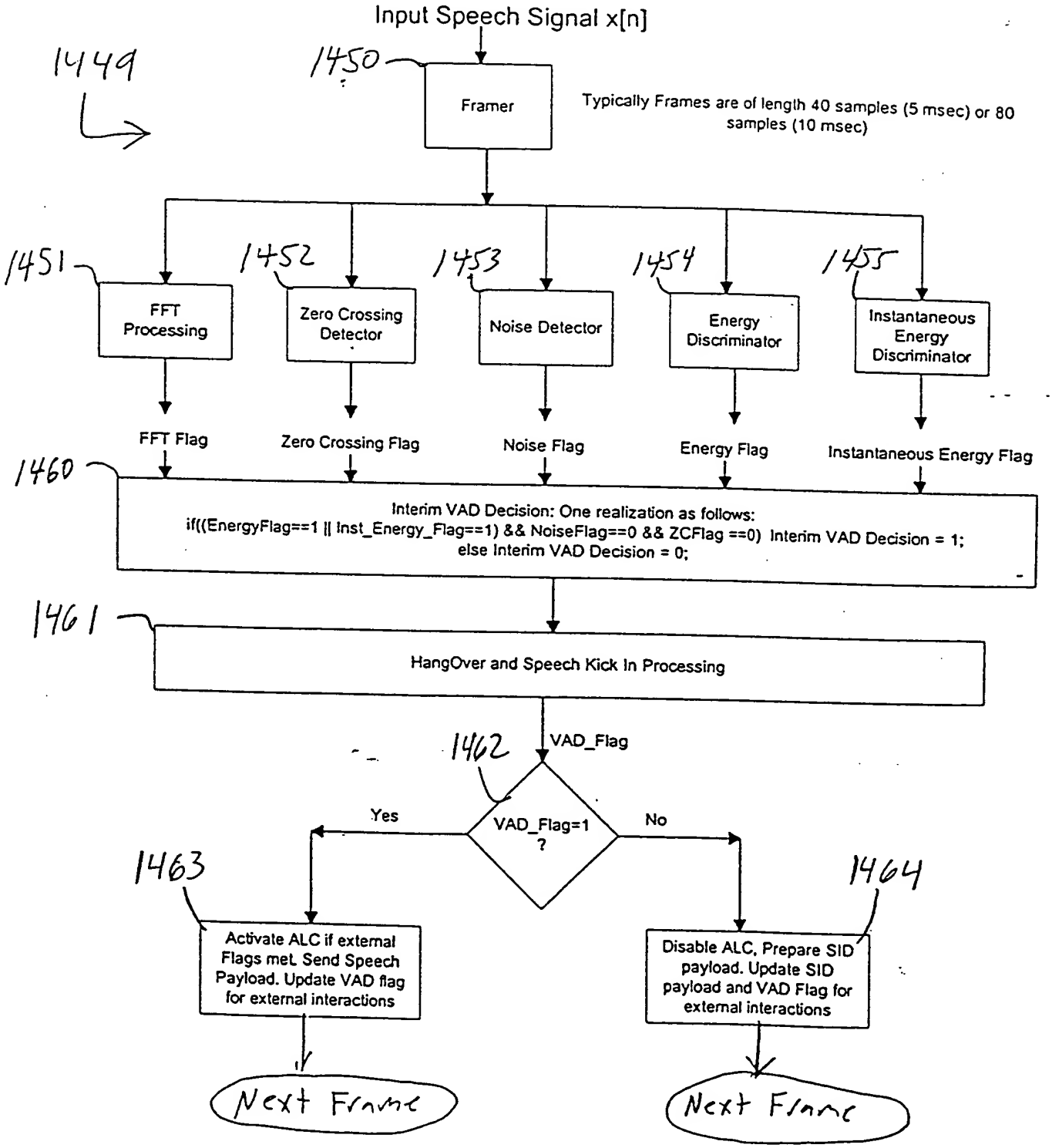


FIG. 14B

FFT Processing of Input Speech for VAD 1451

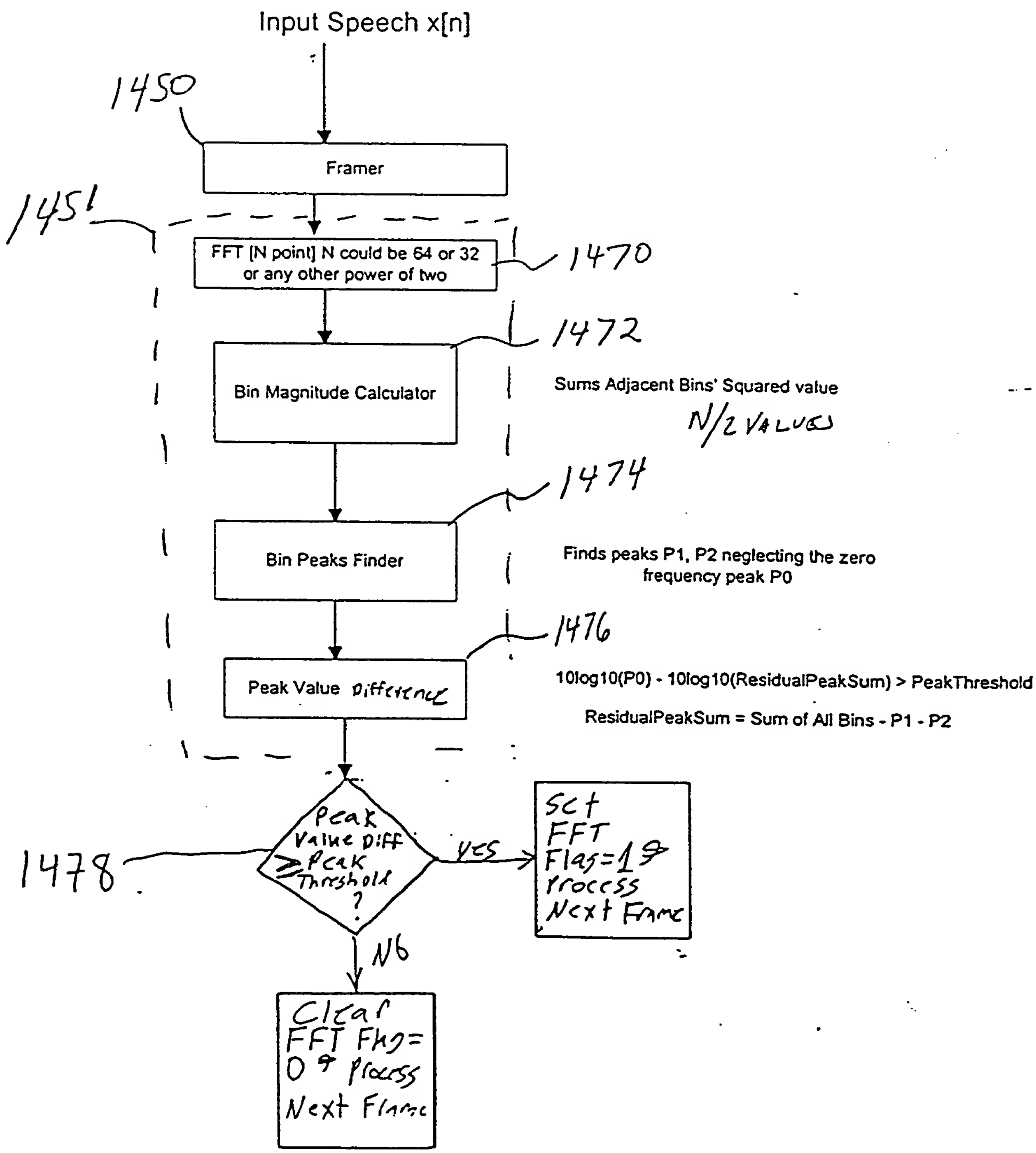


FIG. 14C

Zero Crossing Detector 1452

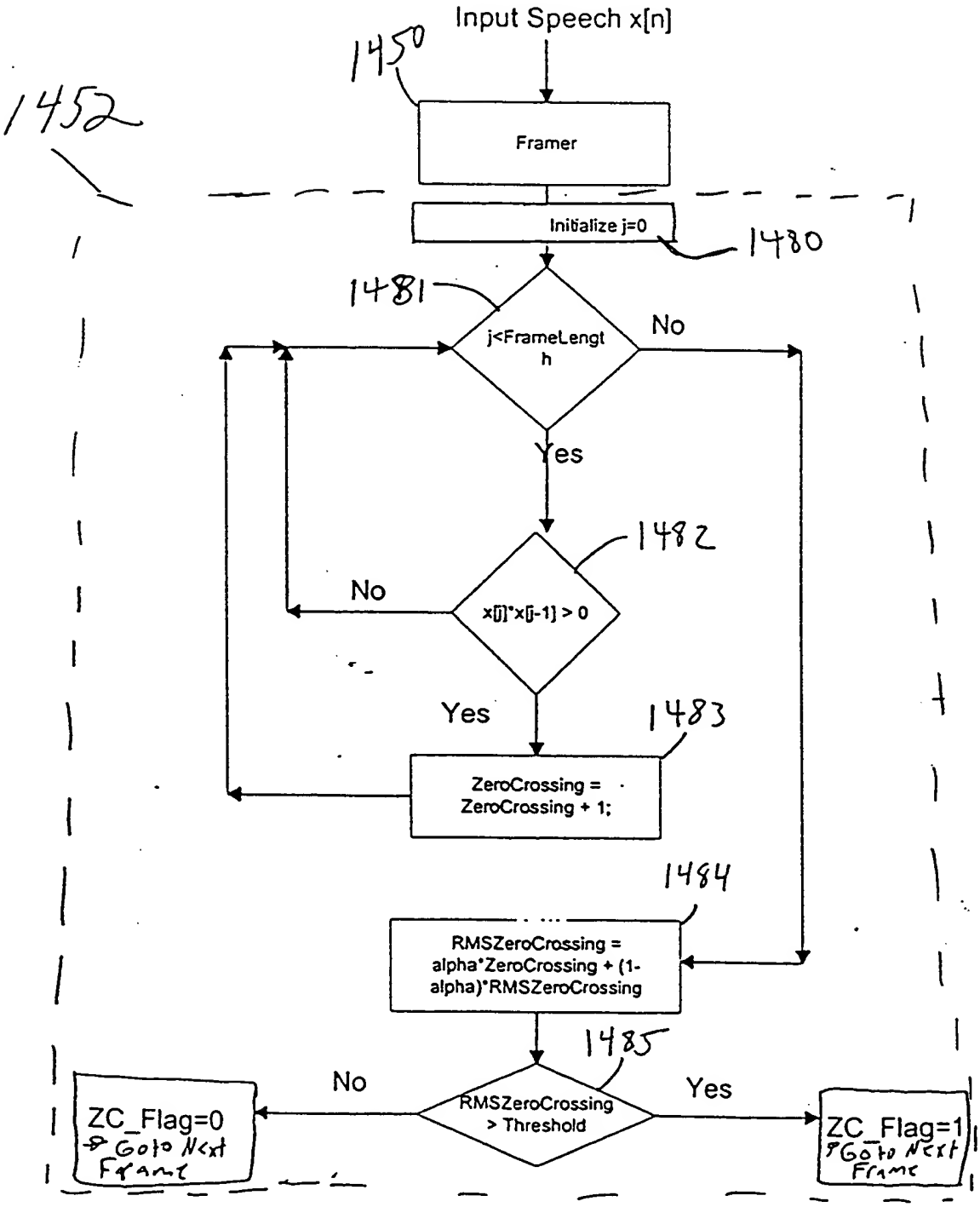


FIG. 14D

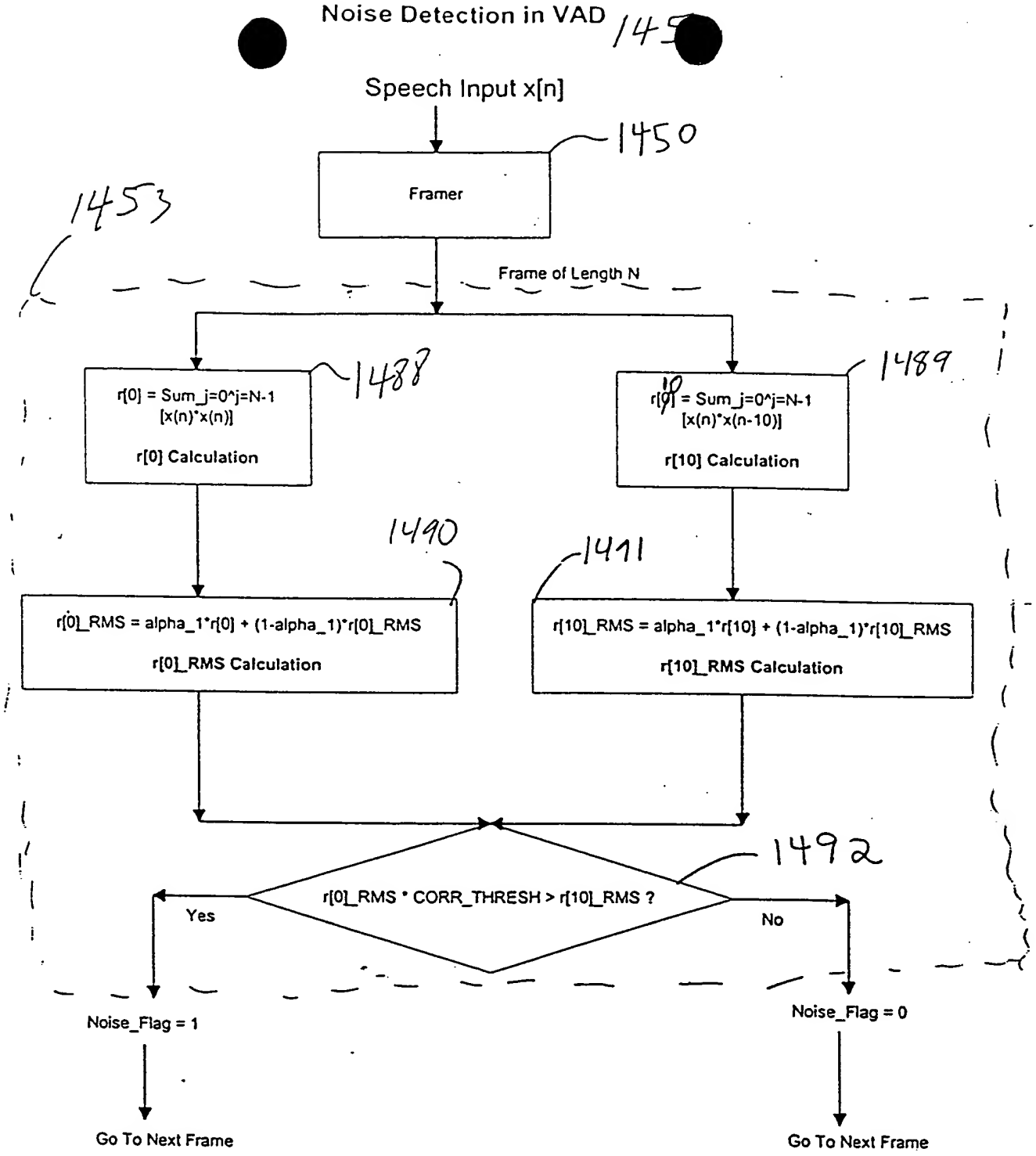

Noise Detection in VAD
1455
Speech Input x[n]
1450
Framer
1453
Frame of Length N
1488
r[0] = Sum_j=0^j=N-1 [x(n)*x(n)]
r[0] Calculation
1489
r[10] = Sum_j=0^j=N-1 [x(n)*x(n-10)]
r[10] Calculation
1490
r[0]_RMS = alpha_1*r[0] + (1-alpha_1)*r[0]_RMS
r[0]_RMS Calculation
1491
r[10]_RMS = alpha_1*r[10] + (1-alpha_1)*r[10]_RMS
r[10]_RMS Calculation
1492
r[0]_RMS * CORR_THRESH > r[10]_RMS ?
Yes
No
Noise_Flag = 1
Noise_Flag = 0
Go To Next Frame
Go To Next Frame

FIG. 14E

Energy Discriminator 1454

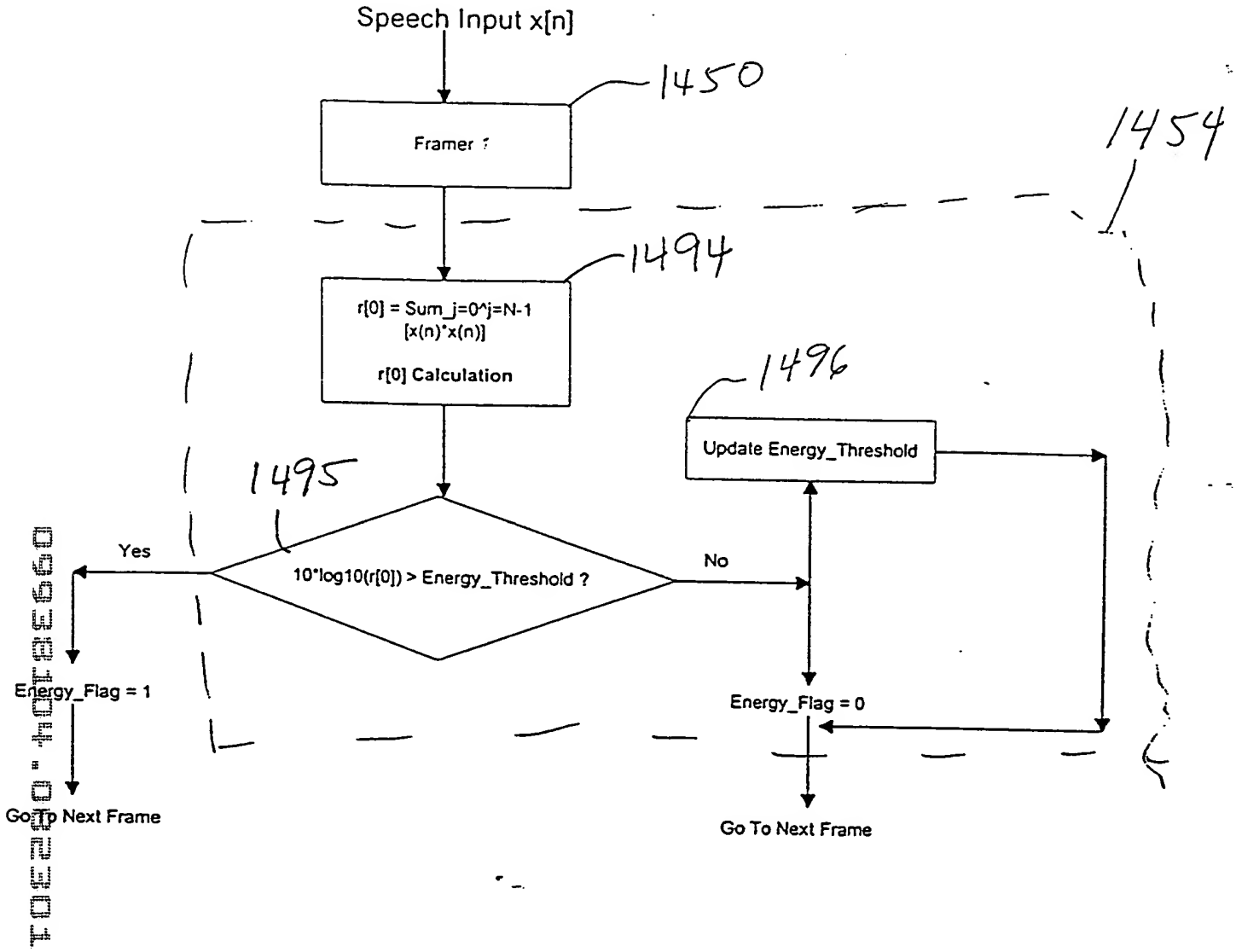


FIG. 14F

Instantaneous Energy Discriminator 1455

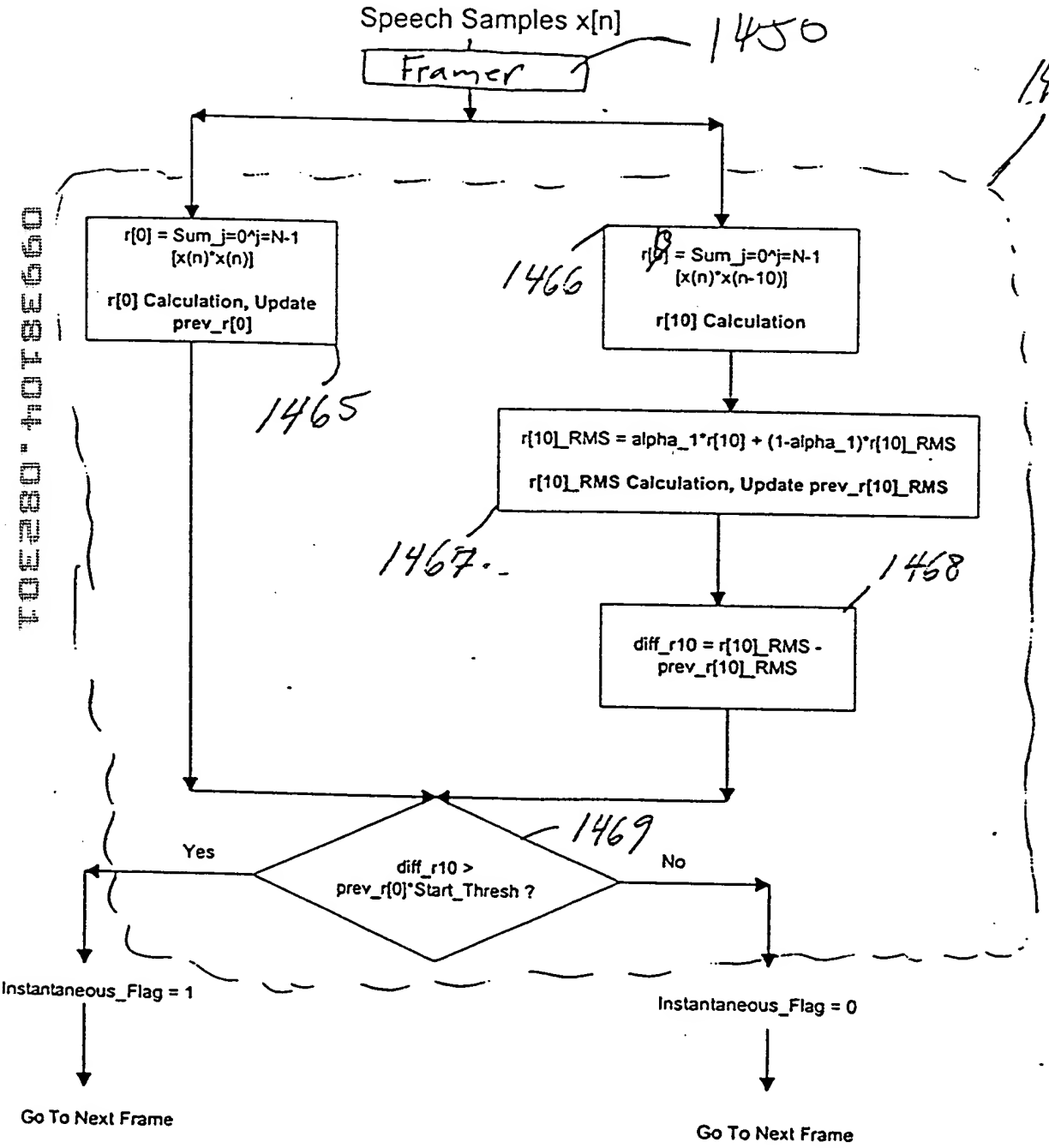


FIG. 14G

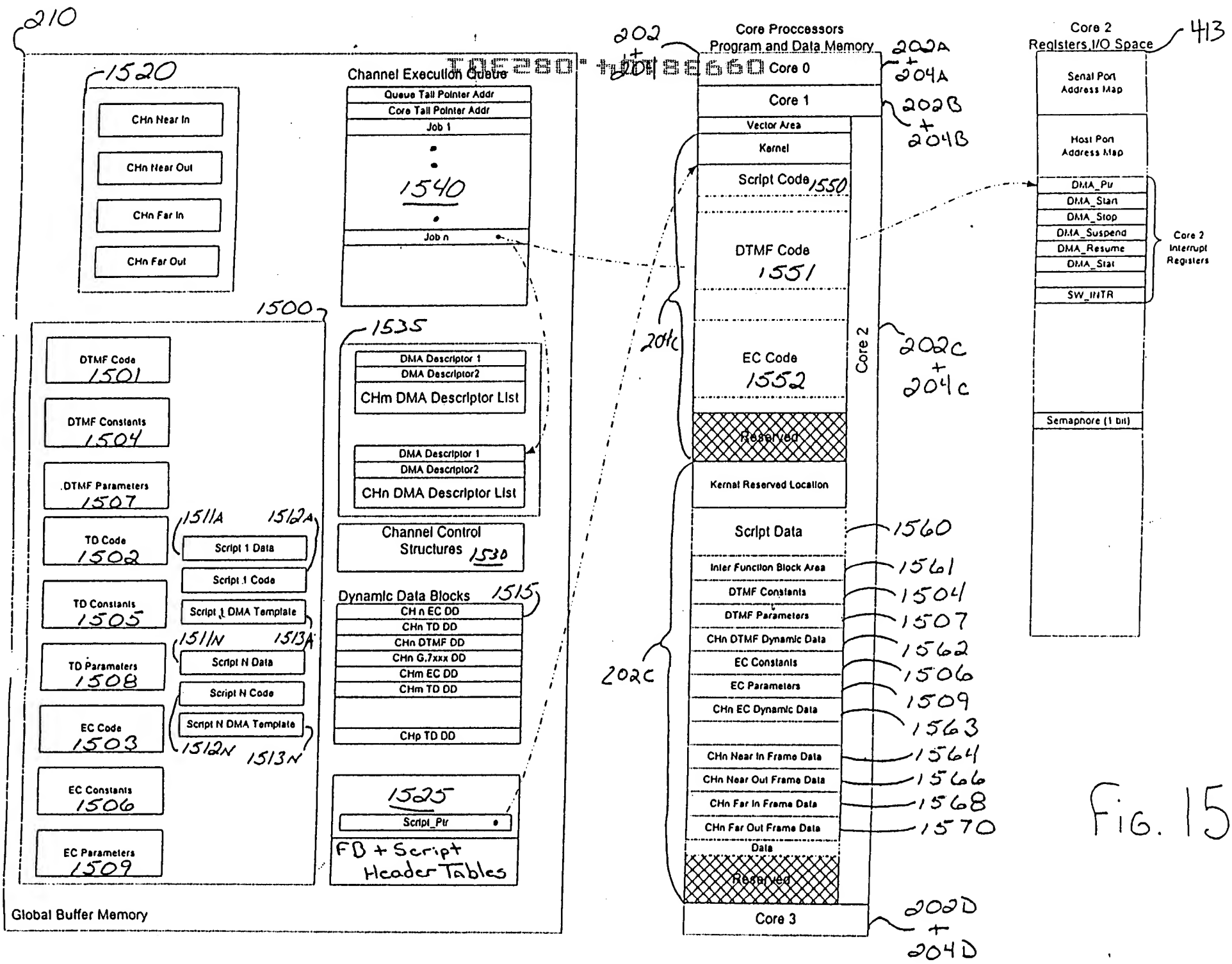

210
1520
CHn Near In
CHn Near Out
CHn Far In
CHn Far Out
Channel Execution Queue
Queue Tail Pointer Addr
Core Tail Pointer Addr
Job 1
1540
Job n
1500
DTMF Code 1501
DTMF Constants 1504
DTMF Parameters 1507
TD Code 1502
TD Constants 1505
TD Parameters 1508
EC Code 1503
EC Constants 1506
EC Parameters 1509
1511A
1512A
Script 1 Data
Script 1 Code
Script 1 DMA Template
1511N
1513A
Script N Data
Script N Code
Script N DMA Template
1512N
1513N
1535
DMA Descriptor 1
DMA Descriptor2
CHm DMA Descriptor List
DMA Descriptor 1
DMA Descriptor2
CHn DMA Descriptor List
Channel Control Structures 1530
Dynamic Data Blocks
1515
CH n EC DD
CHn TD DD
CHn DTMF DD
CHn G.7xxx DD
CHm EC DD
CHm TD DD
CHp TD DD
1525
Script_Ptr
FB + Script Header Tables
Global Buffer Memory
202
Core Proccessors Program and Data Memory
202A
204A
Core 0
Core 1
202B
204B
Vector Area
Kernel
Script Code 1550
DTMF Code 1551
EC Code 1552
Reserved
Kernal Reserved Location
Script Data
1560
Inter Function Block Area
1561
DTMF Constants
1504
DTMF Parameters
1507
CHn DTMF Dynamic Data
1562
EC Constants
1506
EC Parameters
1509
CHn EC Dynamic Data
1563
CHn Near In Frame Data
1564
CHn Near Out Frame Data
1566
CHn Far In Frame Data
1568
CHn Far Out Frame Data
1570
Data
Reserved
Core 3
202D
204D
Core 2
202C
204C
Core 2 Registers I/O Space
413
Serial Port Address Map
Host Port Address Map
DMA_Ptr
DMA_Start
DMA_Stop
DMA_Suspend
DMA_Resume
DMA_Stat
SW_INTR
Core 2 Interrupt Registers
Semaphore (1 bit)

Fig. 15

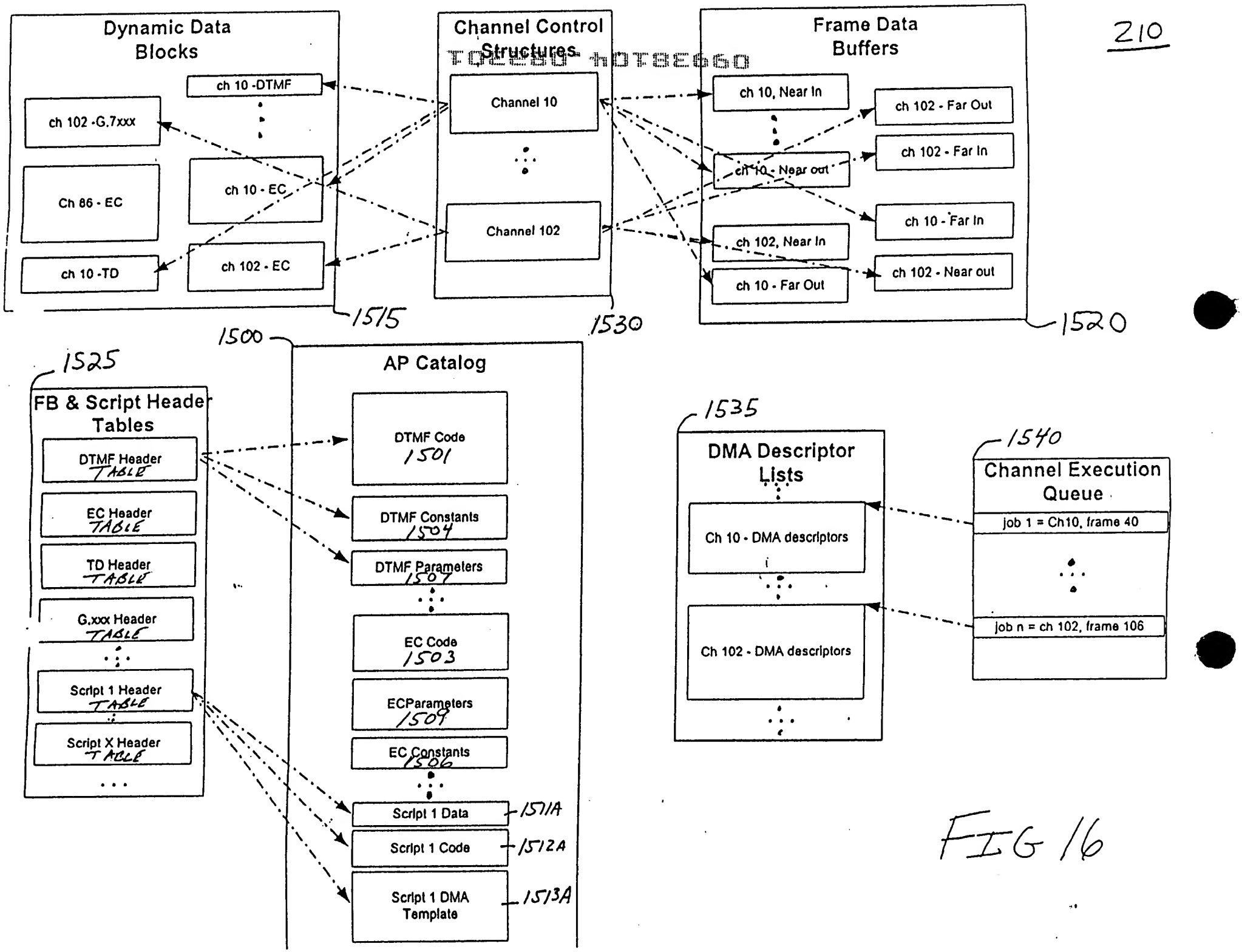

210
Dynamic Data Blocks
ch 10 -DTMF
ch 102 -G.7xxx
Ch 86 - EC
ch 10 - EC
ch 10 -TD
ch 102 - EC
1515
Channel Control Structures
Channel 10
Channel 102
1530
Frame Data Buffers
ch 10, Near In
ch 10 - Near out
ch 102, Near In
ch 10 - Far Out
ch 102 - Far Out
ch 102 - Far In
ch 10 - Far In
ch 102 - Near out
1520
1500
1525
FB & Script Header Tables
DTMF Header TABLE
EC Header TABLE
TD Header TABLE
G.xxx Header TABLE
Script 1 Header TABLE
Script X Header TABLE
AP Catalog
DTMF Code 1501
DTMF Constants 1504
DTMF Parameters 1507
EC Code 1503
ECParameters 1509
EC Constants 1506
Script 1 Data
1511A
Script 1 Code
1512A
Script 1 DMA Template
1513A
1535
DMA Descriptor Lists
Ch 10 - DMA descriptors
Ch 102 - DMA descriptors
1540
Channel Execution Queue
job 1 = Ch10, frame 40
job n = ch 102, frame 106

FIG 16

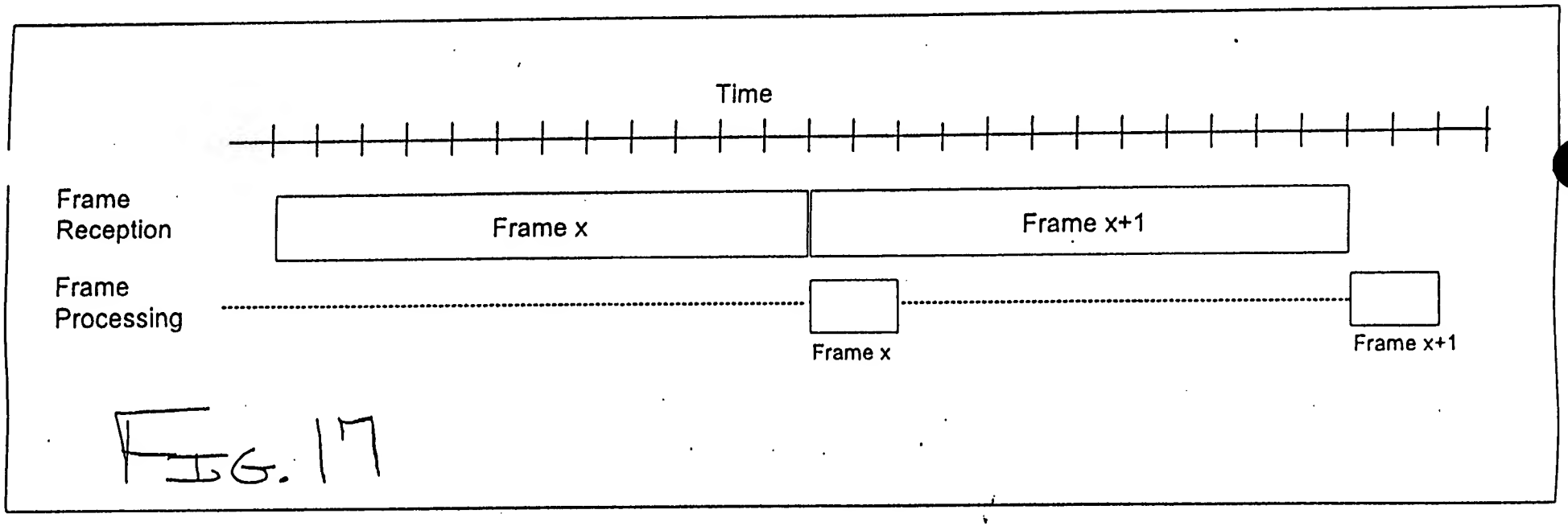


FIG. 17

FIG. 18